FIG. 1

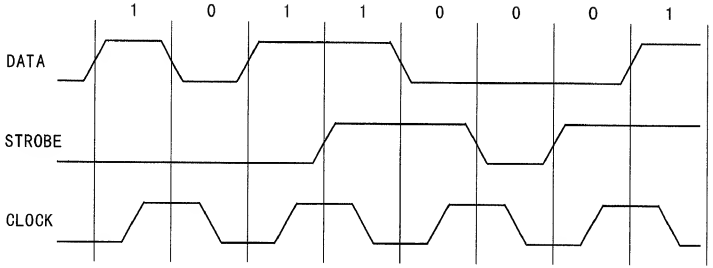


FIG. 2

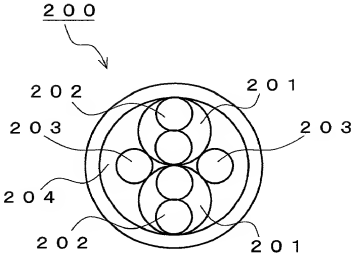

200
202
201
203
203
204
202
201

FIG. 5

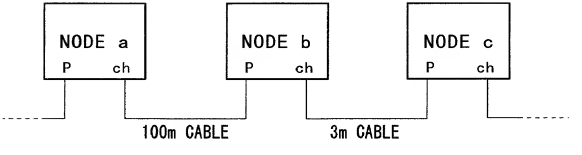

NODE a
P
ch
NODE b
P
ch
NODE c
P
ch
100m CABLE
3m CABLE

FIG. 3A
(BUS INITIALIZATION)

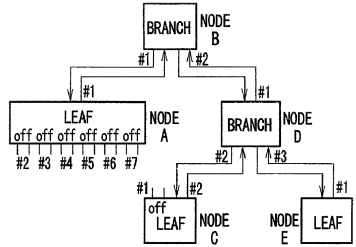


FIG. 3B
(TREE IDENTIFICATION)

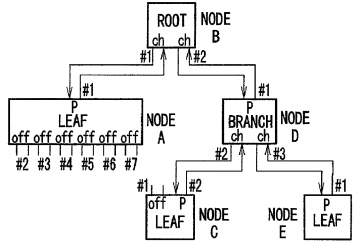


FIG. 3C
(SELF IDENTIFICATION)

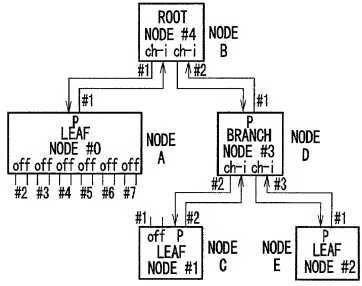
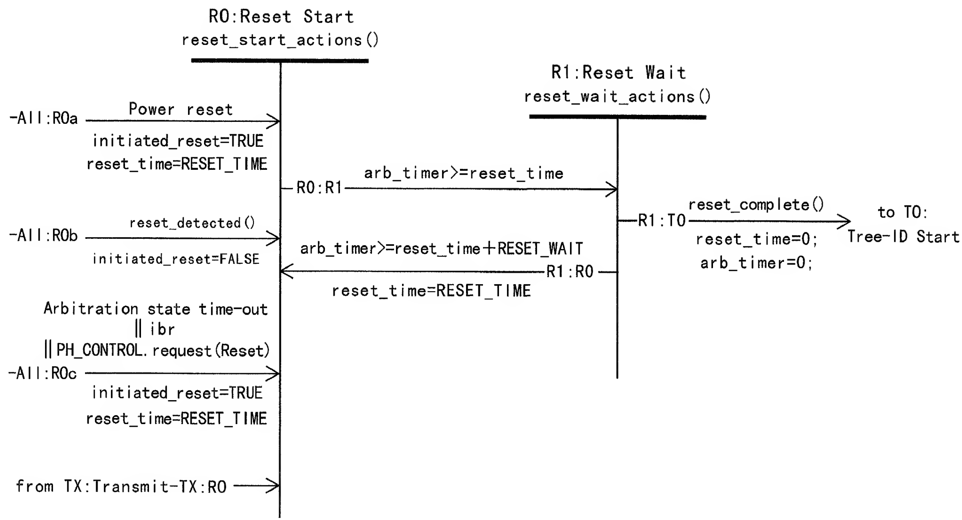
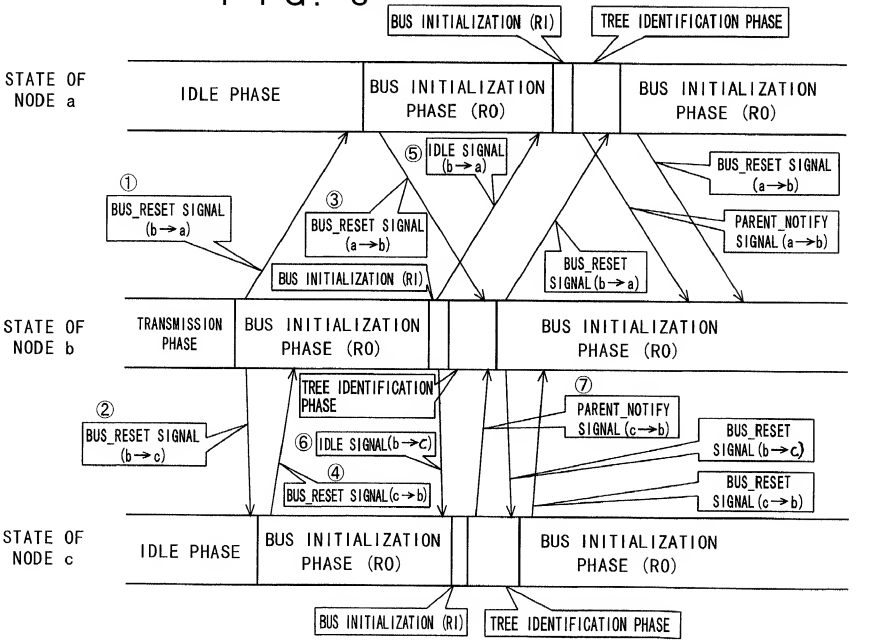


FIG. 4

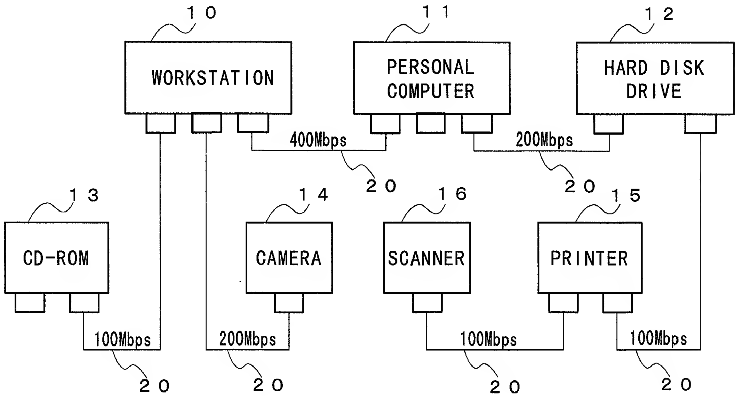


R0:Reset Start
reset_start_actions()
R1:Reset Wait
reset_wait_actions()
-All:R0a
Power reset
initiated_reset=TRUE
reset_time=RESET_TIME
R0:R1
arb_timer>=reset_time
R1:T0
reset_complete()
to T0:
Tree-ID Start
reset_time=0;
arb_timer=0;
-All:R0b
reset_detected()
initiated_reset=FALSE
arb_timer>=reset_time+RESET_WAIT
R1:R0
reset_time=RESET_TIME
Arbitration state time-out
|| ibr
|| PH_CONTROL. request(Reset)
-All:R0c
initiated_reset=TRUE
reset_time=RESET_TIME
from TX:Transmit-TX:R0

FIG. 6



F I G. 7



1 0
WORKSTATION
1 1
PERSONAL COMPUTER
1 2
HARD DISK DRIVE
400Mbps
200Mbps
2 0
2 0
1 3
CD-ROM
1 4
CAMERA
1 6
SCANNER
1 5
PRINTER
100Mbps
200Mbps
100Mbps
100Mbps
2 0
2 0
2 0
2 0

FIG. 8

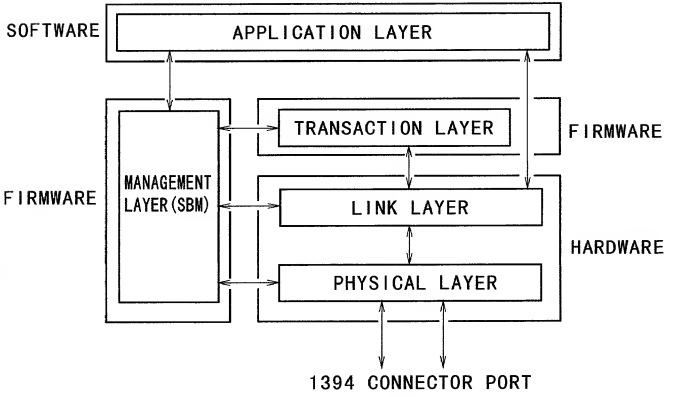

SOFTWARE
APPLICATION LAYER
TRANSACTION LAYER
FIRMWARE
FIRMWARE
MANAGEMENT LAYER(SBM)
LINK LAYER
HARDWARE
PHYSICAL LAYER
1394 CONNECTOR PORT

FIG. 9

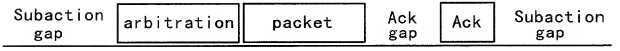

Subaction gap
arbitration
packet
Ack gap
Ack
Subaction gap

FIG. 10A

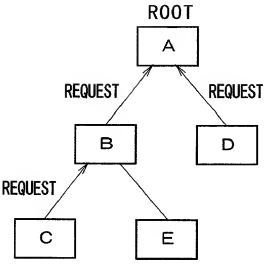


FIG. 10B

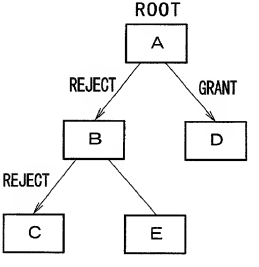


FIG. 11

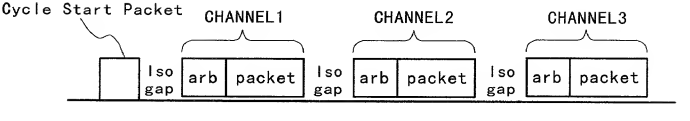
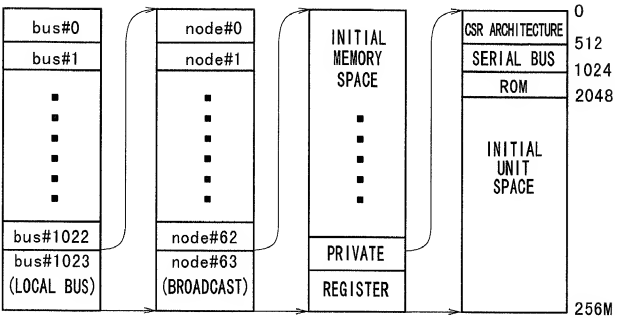


FIG. 12



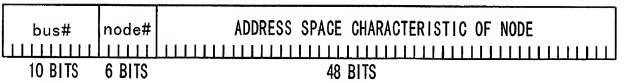


FIG. 13

OFFSETS	NAMES	FUNCTIONS
000h	STATE_CLEAR	STATE AND CONTROL INFORMATION
004h	STATE_SET	SET STATE_CLEAR BIT
008h	NODE_IDs	INDICATE 16-BIT NODE ID
00Ch	RESET_START	START COMMAND RESET
018h-01Ch	SPLIT_TIMEOUT	PRESCRIBE MAXIMUM TIME OF SPLIT
200h	CYCLE_TIME	CYCLE TIME
210h	BUSY_TIMEOUT	PRESCRIBE LIMIT OF RETRY
21Ch	BUS_MANAGER	INDICATE BUS MANAGER ID
220h	BANDWIDTH_AVAILABLE	INDICATE BANDWIDTH THAT CAN BE ASSIGNED TO ISOCHRONOUS COMMUNICATION
224h-228h	CHANNELS_AVAILABLE	INDICATE USED STATE OF EACH CHANNEL

FIG. 14

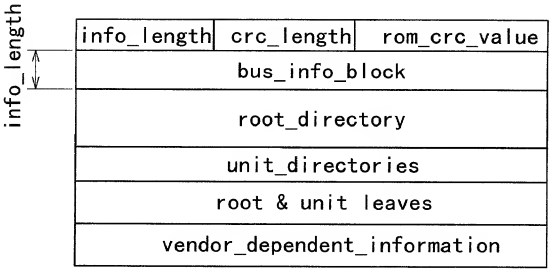


FIG. 16

900h	Output Master Plug Register
904h	Output Plug Control Register #0
908h	Output Plug Control Register #1
⋮	⋮
97Ch	Output Plug Control Register #30
980h	Input Master Plug Register
984h	Input Plug Control Register #0
988h	Input Plug Control Register #1
⋮	⋮
9FCh	Input Plug Control Register #30

FIG. 15

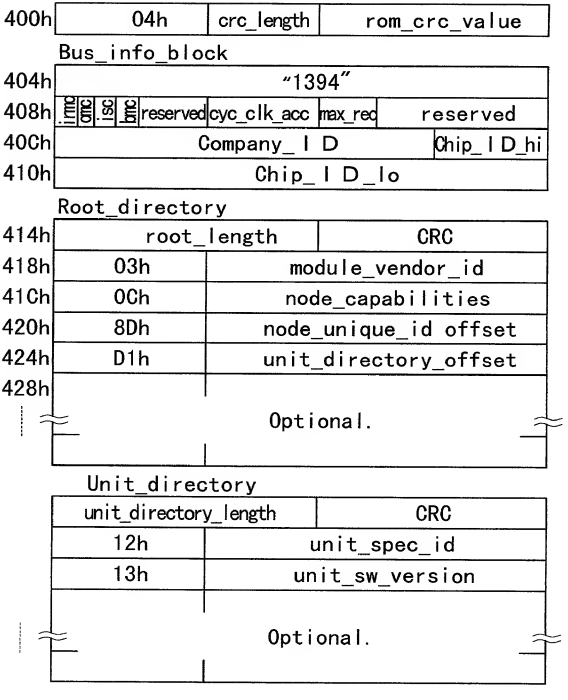

400h
04h
crc_length
rom_crc_value
Bus_info_block
404h
"1394"
408h
irmc
cmc
isc
bmc
reserved
cyc_clk_acc
max_rec
reserved
40Ch
Company_ID
Chip_ID_hi
410h
Chip_ID_lo
Root_directory
414h
root_length
CRC
418h
03h
module_vendor_id
41Ch
0Ch
node_capabilities
420h
8Dh
node_unique_id offset
424h
D1h
unit_directory_offset
428h
Optional.
Unit_directory
unit_directory_length
CRC
12h
unit_spec_id
13h
unit_sw_version
Optional.

FIG. 17A

oMPR

data rate capability	broadcast channel base	non-persistent extension field	persistent extension field	reserved	number of output plugs
2	6	8	8	3	5 (bit)

FIG. 17B

oPCR[n]

on-line	broadcast connection counter	point-to-point connection counter	reserved	channel number	data rate	overhead ID	payload
1	1	6	2	6	2	4	10(bit)

FIG. 17C

iMPR

data rate capability	reserved	non-persistent extension field	persistent extension field	reserved	number of output plugs
2	6	8	8	3	5 (bit)

FIG. 17D

iPCR[n]

on-line	broadcast connection counter	point-to-point connection counter	reserved	channel number	reserved
1	1	6	2	6	16 (bit)

F I G. 1 8

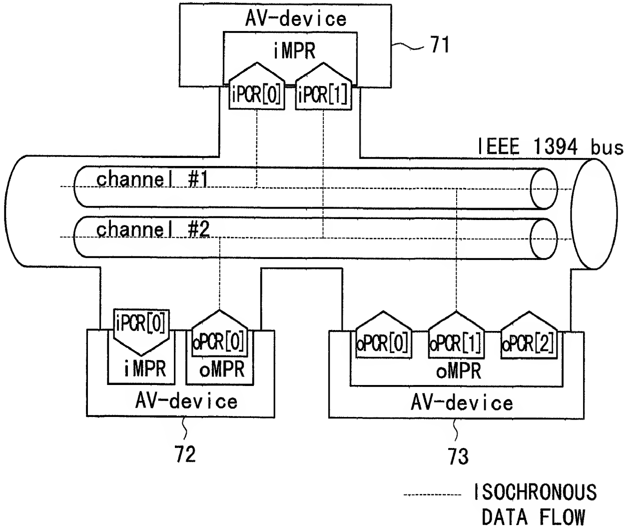

AV-device
iMPR
iPCR[0]
iPCR[1]
71
IEEE 1394 bus
channel #1
channel #2
iPCR[0]
iMPR
oPCR[0]
oMPR
AV-device
72
oPCR[0]
oPCR[1]
oPCR[2]
oMPR
AV-device
73
ISOCHRONOUS
DATA FLOW

FIG. 19

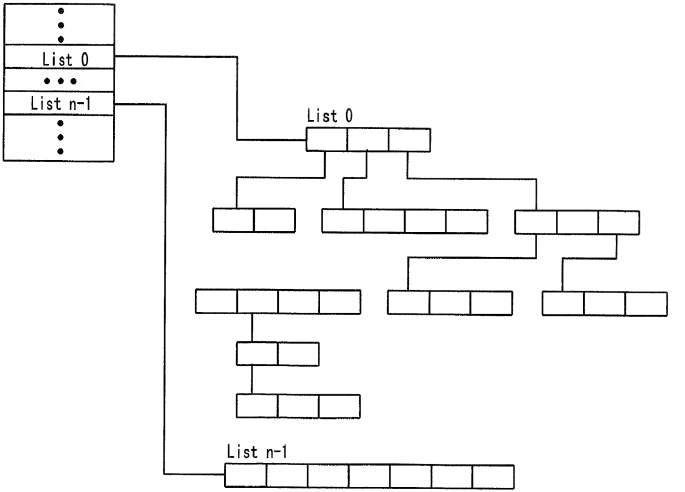


FIG. 20

The General Subunit Identifier Descriptor	
address	contents
00 00_{16}	descriptor_length
00 01_{16}	
00 02_{16}	generation_ID
00 03_{16}	size_of_list_ID
00 04_{16}	size_of_object_ID
00 05_{16}	size_of_object_position
00 06_{16}	number_of_root_object_lists(n)
00 07_{16}	
00 08_{16}	root_object_list_id_0
⋮	
⋮	⋮
⋮	root_object_list_id_n-1
⋮	
⋮	subunit_dependent_length
⋮	
⋮	subunit_dependent_information
⋮	
⋮	
⋮	manufacturer_dependent_length
⋮	
⋮	manufacturer_dependent_information
⋮	
⋮	

FIG. 21

generation_ID values	
generation_ID	meaning
00_{16}	Data structures and command sets as specified in the AV/C General Specification, version 3.0
all others	reserved for future specification

FIG. 22

List ID Value Assignment Ranges	
range of values	list definition
0000_{16}-$0FFF_{16}$	reserved
1000_{16}-$3FFF_{16}$	subunit-type dependent
4000_{16}-$FFFF_{16}$	reserved
1 0000_{16}-max list ID value	subunit-type dependent

FIG. 23

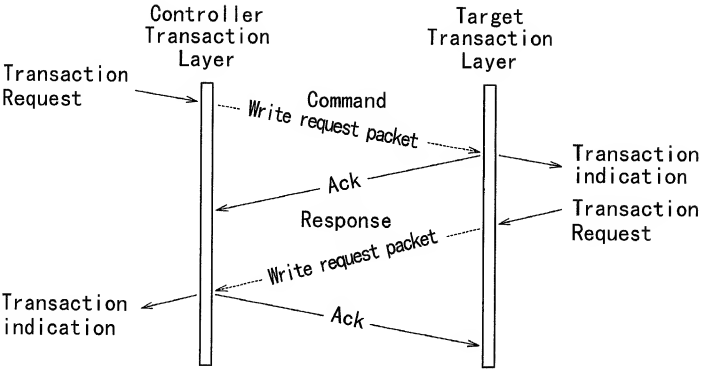


FIG. 24

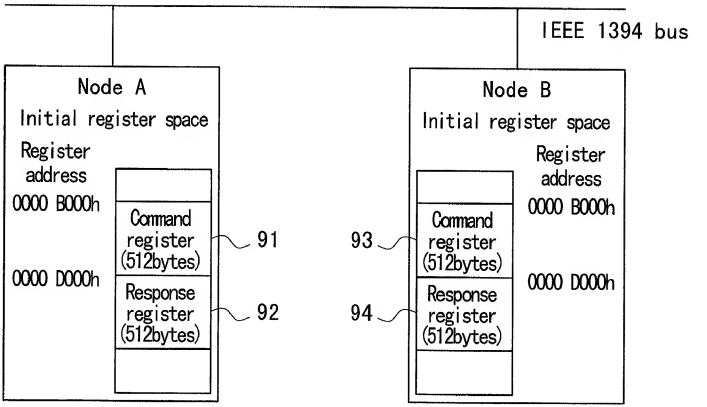
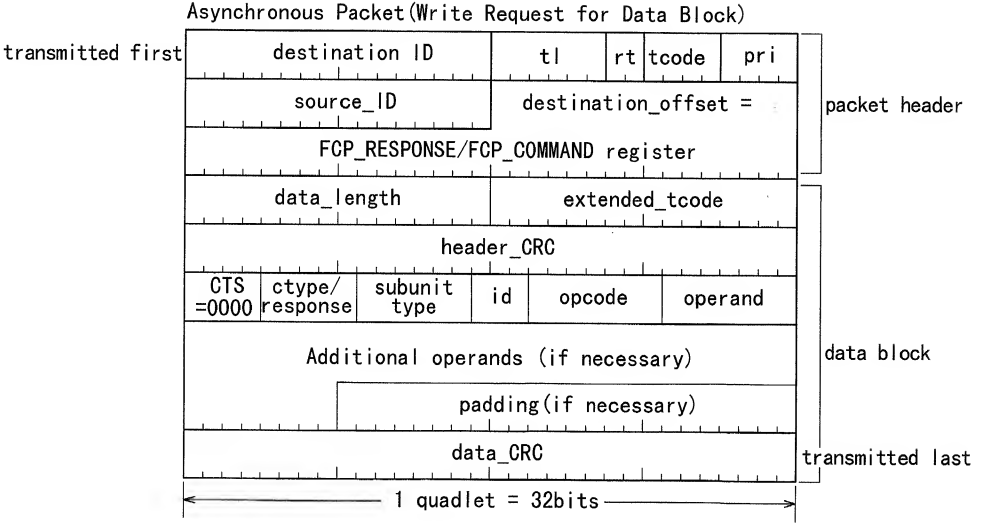


FIG. 25


Asynchronous Packet(Write Request for Data Block)
transmitted first
destination ID
tl
rt
tcode
pri
source_ID
destination_offset =
FCP_RESPONSE/FCP_COMMAND register
packet header
data_length
extended_tcode
header_CRC
CTS =0000
ctype/ response
subunit type
id
opcode
operand
Additional operands (if necessary)
padding(if necessary)
data block
data_CRC
transmitted last
1 quadlet = 32bits

ctype/response

Command	0000	CONTROL
	0001	STATUS
	0010	SPECIFIC INQUIRY
	0011	NOTIFY
	0100	GENERAL INQUIRY
	0101 ⟨ 0111	(reserved for future specification)
Response	1000	NOT IMPLEMENTED
	1001	ACCEPTED
	1010	REJECTED
	1011	IN TRANSITION
	1100	IMPLEMENTED/STABLE
	1101	CHANGED
	1110	(reserved for future specification)
	1111	INTERIM

FIG. 26A

subunit_type

00000	Video monitor
⟨	(reserved)
00011	Disc recorder/ Player
00100	Tape recorder/ Player
00101	Tuner
00111	Video Camera
⟨	(reserved)
11100	Vendor unique
11101	reserved
11110	Subunit type extended to next byte
11111	Unit

FIG. 26B

opcode:Operation Code

00h	VENDOR-DEPENDENT
50h	SEARCH MODE
51h	TIMECODE
52h	ATN
60h	OPEN MIC
61h	READ MIC
62h	WRITE MIC
C1h	LOAD MEDIUM
C2h	RECORD
C3h	PLAY
C4h	WIND
⟨	⟨

FIG. 26C

FIG. 27A

AV/C	control	tape recorder /player	IN THE CASE OF ID0	PLAY	FORWARD
CTS= 0000	ctype= 0000	subunit type= 00100	id= 000	opcode= C3h	operand= 75h

FIG. 27B

AV/C	accepted	tape recorder /player	IN THE CASE OF ID0	PLAY	FORWARD
CTS= 0000	response =1001	subunit type= 00100	id= 000	opcode= C3h	operand= 75h

F I G. 2 8

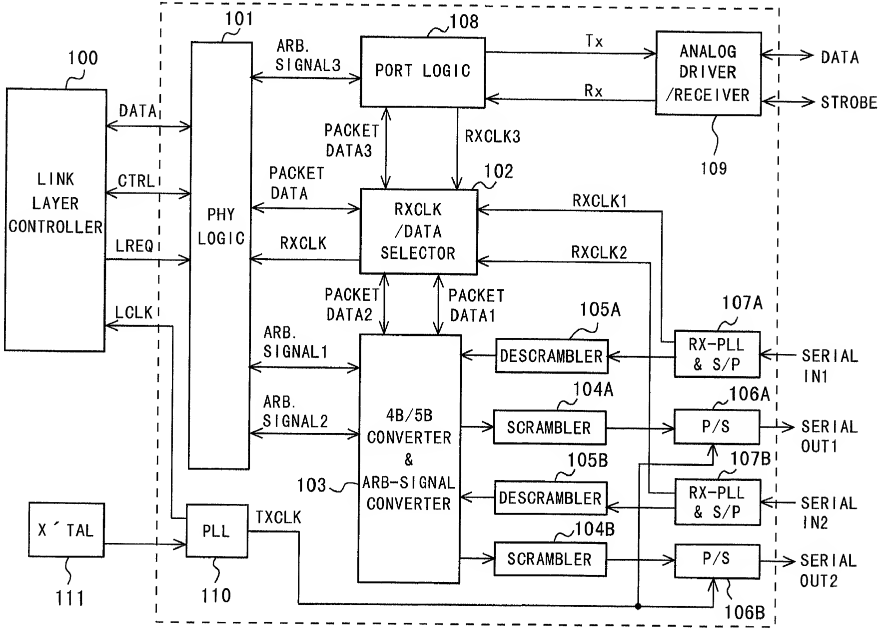
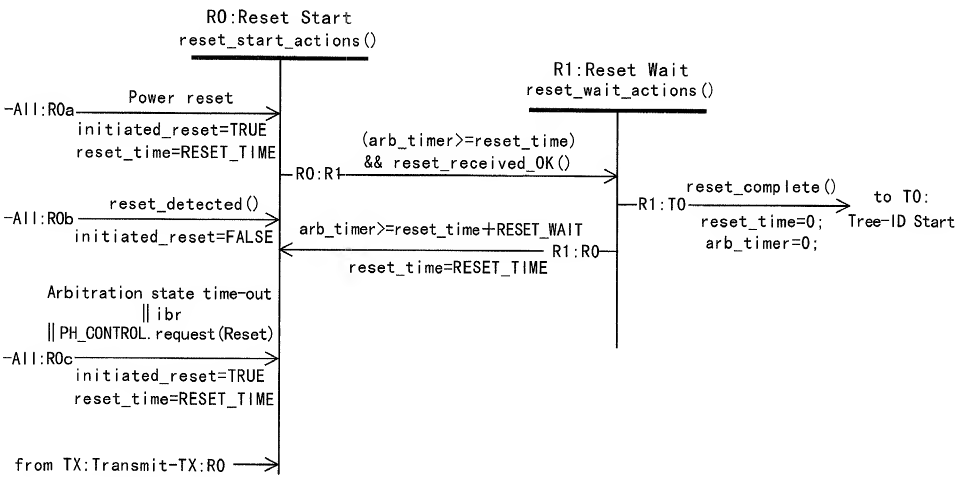

100
LINK LAYER CONTROLLER
DATA
CTRL
LREQ
LCLK
101
PHY LOGIC
ARB. SIGNAL3
PACKET DATA
RXCLK
ARB. SIGNAL1
ARB. SIGNAL2
108
PORT LOGIC
Tx
Rx
ANALOG DRIVER /RECEIVER
109
DATA
STROBE
PACKET DATA3
RXCLK3
102
RXCLK /DATA SELECTOR
RXCLK1
RXCLK2
PACKET DATA2
PACKET DATA1
103
4B/5B CONVERTER & ARB-SIGNAL CONVERTER
105A
DESCRAMBLER
104A
SCRAMBLER
105B
DESCRAMBLER
104B
SCRAMBLER
107A
RX-PLL & S/P
106A
P/S
107B
RX-PLL & S/P
106B
P/S
SERIAL IN1
SERIAL OUT1
SERIAL IN2
SERIAL OUT2
X´TAL
111
PLL
110
TXCLK

FIG. 29



R0:Reset Start
reset_start_actions()
R1:Reset Wait
reset_wait_actions()
-All:R0a
Power reset
initiated_reset=TRUE
reset_time=RESET_TIME
R0:R1
(arb_timer>=reset_time)
&& reset_received_OK()
R1:T0
reset_complete()
reset_time=0;
arb_timer=0;
to T0:
Tree-ID Start
-All:R0b
reset_detected()
initiated_reset=FALSE
arb_timer>=reset_time+RESET_WAIT
R1:R0
reset_time=RESET_TIME
Arbitration state time-out
|| ibr
|| PH_CONTROL.request(Reset)
-All:R0c
initiated_reset=TRUE
reset_time=RESET_TIME
from TX:Transmit-TX:R0

FIG. 30